Discrete Mathematics 115 (1993) 187–199
North-Holland

Transfer flow graphs

Klaus Jansen
FBIV-Mathematics, University of Trier, Postfach 3825, 5500 Trier, Germany

Received 17 October 1990
Revised 30 July 1991

Abstract

Jansen, K., Transfer flow graphs, Discrete Mathematics 115 (1993) 187–199.

We consider four combinatorial optimization problems (independent set, clique, coloring, partition into cliques) for a special graph class. These graphs $G=(N,E)$ are generated by nondisjoint union of a cograph and a graph with m disjoint cliques. Such graphs can be described as transfer flow graphs. At first, we show that these graphs occur as compatibility graphs in the synthesis of hardware configurations. Then we prove that the clique problem can be solved in $O(|N|^2)$ steps and that the other problems are NP-complete. Moreover, all four problems can be solved in polynomial time if m is constant.

1. Introduction

One of the important problems in the combinatorial graph optimization is the problem of finding a maximum independent set or a maximum clique and the problem of finding a minimum coloring or a minimum partition into cliques, respectively. We denote these sizes for a graph G with $\alpha(G), \omega(G), \chi(G)$ and $\kappa(G)$. In a historical paper, Karp [8] has shown that these optimization problems are NP-complete for general graphs. Since often only a subclass of all graphs occurs, it is important to know how difficult the subproblems are. Johnson [7] has summarized the results for several graph classes. For example, the four problems are polynomial solvable for perfect graphs. A graph is perfect if, for each induced subgraph G' of G, $\chi(G')=\omega(G')$. Using the ellipsoid method, Grötschel et al. [4] have given algorithms for these problems. Simpler and combinatorial methods are known for subclasses of the perfect graphs [1, 3, 5, 9].

For cographs, graphs without a path of length 4 as induced subgraph, there exist linear algorithms. These follow from the facts that cographs can be constructed with two operations on disjoint graphs and that the algorithms can be applied recursively.

Correspondence to: Klaus Jansen, FBIV-Mathematics, University of Trier, Postfach 3825, 5500 Trier, Germany.

If $G_1=(N_1,E_1)$ and $G_2=(N_2,E_2)$ are graphs, $\bigcup(G_1,G_2)=(N_1\cup N_2,E_1\cup E_2)$ is the union and $+(G_1,G_2)=(N_1\cup N_2,E_1\cup E_2\cup\{\{n_1,n_2\}\mid n_i\in N_i\})$ is the join of G_1 and G_2. The tree which corresponds to these operations for a given cograph is called a cotree. For example, the chromatic number $\chi(G)$ for a cograph [10] can be computed as follows:

$$\chi(G)=\begin{cases}\max\{\chi(G_1),\chi(G_2)\}, & G=\bigcup(G_1,G_2),\\ \chi(G_1)+\chi(G_2), & G=+(G_1,G_2),\\ 1, & G=(\{n\},\emptyset).\end{cases}$$

Similar calculations exist for the other problems. A subclass of the cographs is given by disjoint union of complete graphs. We call these graphs clique-partition graphs, because we can define for these graphs a unique mapping $L:N\to\{1,\dots,m\}$ which gives a minimum partition into cliques. Let $Z_m=\{1,\dots,m\}$. Therefore, we examine labelled cographs (G,L), where $G=(N,E)$ is a cograph and $L:N\to Z_m$ is a mapping. It is allowed to use a mapping $L:\bar{N}\to Z_m$ with $N\subset\bar{N}$. The optimization problems which we examine correspond to the following graph:

$$G^L=(N,E\cup\{\{u,v\}\mid u,v\in N,\ u\neq v,\ L(u)=L(v)\}),$$

which we call transfer flow graphs; this name will be motivated in the next section. These graphs can be seen as a superposition of two graphs of equal cardinality, one being a cograph and the other a disjoint union of cliques.

Let $K(U,G)=\{x\in N\mid L(x)\in U\}$ be the set of nodes of a cograph $G=(N,E)$ with labelling $U\subset Z_m$ and let $L(V)=\{L(v)\mid v\in V\}$ be the set of labels of the nodes $V\subset N$. The subgraph of G induced by a set $V\subset N$ is denoted by $G|_V$. The cographs and the transfer flow graphs are closed under the induced subgraph operation. But transfer flow graphs are not necessarily perfect; see, for example, a cycle of length five.

In Section 2 we show that the transfer flow graphs occur as compatibility graphs in the design of hardware structures. They represent a bus assignment problem. After that, we examine the four optimization problems for these graphs with general and constant number m of labels.

2. Characterization

In this section we describe a problem in which the transfer flow graphs occur as compatibility graphs. At first, we define the model of a flow digraph. Similar structures are used in compiler design; see e.g. [6].

Definition 2.1. A flow digraph $\mathscr{F}=(D,Op,F,J,b)$ is an acyclic digraph $D=(N_{\mathscr{F}},E_{\mathscr{F}})$ with nodes $N_{\mathscr{F}}=\{s,t\}\cup Op\cup F\cup J$, edges E, sets F,J, with $|F|=|J|=m\in\mathbb{N}_0$, and a weight function $b:\{(F_i,x)\mid(F_i,x)\in E_{\mathscr{F}},F_i\in F\}\to\{0,1\}$ defined on the edges, such that the following conditions hold:

(1) For each node $x \in N_{\mathscr{F}}$, there exists a directed path from s over x to t.

(2) For each node $F_i \in F$, there exist nodes $x, x' \in N_{\mathscr{F}}$, with $x \neq x'$, and edges $(F_i, x), (F_i, x') \in E_{\mathscr{F}}$, with $b((F_i, x)) \neq b((F_i, x'))$. Further, for each such pair x, x' of nodes, the first common successor of x and x' is the node $J_i \in J$.

(3) For all pairs (F_i, J_i), it holds that all directed paths from F_i to t go over J_i and that all directed paths from s to J_i contain F_i.

We call the elements of Op the operations, a node F_i a fork node, J_i a join node, s the start and t the terminal node of the flow digraph $\mathscr{F}$. The set $N(F_i, J_i) = \{x \in N_{\mathscr{F}} \mid \text{there is a path from } F_i \text{ over } x \text{ to } J_i\} \backslash \{F_i, J_i\}$ can be divided into two disjoint sets $N(F_i, J_i)_0$ and $N(F_i, J_i)_1$. These are the sets of nodes which can be reached over a zero- or one-valuated edge, respectively, from the node F_i. A flow digraph is a loop-free digraph with branching nodes F_i, J_i. Depending upon the control function $\psi: F \to \{0, 1\}$ of the branching nodes, either the operations in $N(F_i, J_i)_0$ or the operations in $N(F_i, J_i)_1$ are executed. For a control function $\psi: F \to \{0, 1\}$, the set of executed operations for ψ is defined by

$$O_\psi = Op \backslash \bigcup_{1 \leqslant i \leqslant m} N(F_i, J_i)_{1-\psi(F_i)}.$$

If the operations are transfers from one module to another, we need a finite nonempty set M of modules and a mapping typ, with

$$typ: Op \to M \times M \backslash \{(x, x) \mid x \in M\}.$$

For an operation $op \in Op$, the mapping $typ(op) = (x_1, x_2)$ indicates that we have a transfer from module x_1 to module x_2. We call x_1 the source and x_2 the sink of the transfer operation op. If, for a discrete set of times $Z_k = \{1, \ldots, k\}$, there is a time table $T: Op \to Z_k$ with the following property:

> for each control function $\psi: F \to \{0, 1\}$ and for all operations $op, op' \in O_\psi$ with directed path from op to op', $T(op) < T(op')$,

we obtain, with $(\mathscr{F}, M, typ, T)$, a transfer digraph. Two operations $op, op' \in Op$, with $typ(op) = (x_1, x_2)$ and $typ(op') = (x'_1, x'_2)$, of a transfer digraph are compatible $(op \sim op')$ iff

(1) $T(op) \neq T(op')$ or

(2) there is no control function $\psi: F \to \{0, 1\}$ with $\{op, op'\} \subset O_\psi$ or

(3) $x_1 = x'_1$.

This means that the operations are incompatible if they are executed in parallel for a control function ψ and if their sources are different. The compatibility graph is defined by $K = (Op, E_O)$ with edges $E_O = \{\{op, op'\} \mid op \sim op'\}$. The graph K_z, the so-called local compatibility graph for time step $z \in Z_m$, is the subgraph of K induced by the set $OP_z = \{op \in Op \mid z = T(op)\}$. We get the whole compatibility graph K by disjoint join operation on the local compatibility graphs K_z.

Our goal is to determine the minimum number of busses; that is, an assignment of the transfers to different busses, also called interconnecting channels. On these channels, informations from one module to another can be transported. But only compatible transfers can use the same channel. For flow digraphs without branchings, this problem is examined by Torng and Wilhelm [11] and by Tseng and Siewiorek [12]. The general problem can be formulated as follows.

Problem: Bus assignment.
Instance: A transfer digraph $(\mathscr{F}, M, typ, T)$ and a number $R \in \mathbb{N}$.
Question: Is there a mapping $f: Op \to Z_R$ with $f(op) \neq f(op')$ for all incompatible pairs of transfers op, op'?

This problem corresponds to a partition of the operations Op in cliques $C_1, \ldots, C_R$. The compatibility graph can be determined simply, because it can be represented as a transfer flow graph.

Theorem 2.2. *Let $(\mathscr{F}, M, typ, T)$ be a transfer digraph with corresponding compatibility graph $K = (Op, E_O)$. Then K is a union of a cograph and a clique-partition graph.*

Proof. We consider the graphs P_z, for a time step $z \in Z_m$, which arise from the second condition of the compatibility relation. This condition is equivalent, for operations $op, op' \in Op_z$, to the following statement:

> The first common predecessor of op and op' is a fork node F_i with $op \in N(F_i, J_i)_0$ and $op' \in N(F_i, J_i)_1$ or vice versa.

Since the flow digraphs are hierarchically structured by the fork–join blocks $N(F_i, J_i)$, we get the graphs P_z by disjoint union and join operations on graphs, starting with a single-vertex graph. Therefore, the graphs P_z are cographs. Since the cographs are closed under the join operation, the graph arising from the first and second condition is a cograph, too. The graph from the third condition is a clique-partition graph. This implies the assertion of the theorem. □

Now we look at the other direction and get a characterization for the class of all compatibility graphs.

Theorem 2.3. *For each transfer flow graph G^L, we can construct a transfer digraph such that the corresponding compatibility graph is equal to G^L.*

Proof. Let G^L be the union of a cograph $G = (N, E)$ and a clique-partition graph with equal set of nodes N and let $L: N \to Z_m$ be the corresponding labelling. Recursively, on the cotree representation of the cograph we can construct the flow digraph $\mathscr{F}$ in a simple manner where $Op = N$ is the set of operations and where each operation can be executed at the same time slot $T(op) = 1$ depending only on the control functions.

As a set of modules, we can choose $M = Z_m \cup N$ and, as transfer mapping *typ*, we define $typ(op) = (L(op), op)$. Then the corresponding compatibility graph is equal to G^L. □

The presentation of the transfer flow graph for the compatibility graph can be computed in $O(|N_{\mathscr{F}}| + |E_{\mathscr{F}}|)$ time by depth-first search in the flow digraph.

3. Clique

Next, we want to solve the clique problem recursively. The following lemma describes the recursive structure of the cliques in the transfer flow graph.

Lemma 3.1. *Let G^L be a transfer flow graph, where $G = (N, E) = op(G_1, G_2)$ and $G_i = (N_i, E_i)$ are cographs and $L: N \to Z_m$ is a labelling.*
If $op = +$, we have

$$\omega(G^L) = \omega(G_1^L) + \omega(G_2^L).$$

If $op = \cup$, we have

$$\omega(G^L) = \max\{\omega(G_1^L), \omega(G_2^L), |K(\{1\} G)|, \dots, |K(\{m\}, G)|\}.$$

Proof. Let C be a clique of $+(G_1, G_2)^L$. The clique C can be divided into two cliques C_1, C_2, where $C_i = C \cap N_i$ is a clique of G_i^L. Conversely, let C_i, for $i \in \{1, 2\}$, be cliques of G_i^L. Then $C_1 \cup C_2$ is a clique of G^L.

For the other case, let C be a clique of $\bigcup(G_1, G_2)^L$. Either C lies in one of the graphs G_i^L, or the clique C can be partitioned into two nonempty sets C_1, C_2, where C_i is a clique of G_i^L. In the second case, the labels for all pairs $u \in C_1$ and $v \in C_2$ are equal: $L(u) = L(v)$. Therefore, $|L(C)| = 1$. Conversely, each clique C of G_i^L and each set $K(\{k\}, G)$ is a clique of G^L, too. □

Theorem 3.2. *The maximum-clique problem can be solved in $O(|N|m) = O(|N|^2)$ steps for a transfer flow graph G^L with cograph $G = (N, E)$ and m labels.*

Corollary 3.3. *Let $(\mathscr{F}, M, typ, T)$ be a transfer digraph with a set Op of transfer operations. The maximum compatible set of transfer operations can be determined in $O(|Op|^2 + |N_{\mathscr{F}}| + |E_{\mathscr{F}}|)$ steps, where $N_{\mathscr{F}}$ are the nodes and $E_{\mathscr{F}}$ are the edges of the flow digraph $\mathscr{F}$.*

4. Independent set

Similarly as for the clique problem, a maximum independent set can be computed recursively for a transfer flow graph. But the computation is more difficult than in the first case.

Lemma 4.1. *Let G^L be a transfer flow graph, where $G=(N,E)=op(G_1,G_2)$ and $G_i=(N_i,E_i)$ are cographs and $L:N\to Z_m$ is a labelling.*

If $op=+$, then

$$\alpha(G^L)=\max\{\alpha(G_1^L),\alpha(G_2^L)\}.$$

If $op=\cup$, then

$$\alpha(G^L)=\max_{\emptyset\subset H\subset Z_m}\{\alpha(G_1^L|_{K(H,G)})+\alpha(G_2^L|_{K(Z_m\setminus H,G)})\}.$$

Proof. Let U be an independent set of $G^L=+(G_1,G_2)^L$. Then the set U is either an independent set of G_1^L or an independent set of G_2^L. Conversely, each independent set of G_i^L is an independent set of G^L, too.

If U is an independent set of $G^L=\bigcup(G_1,G_2)^L$, either U is an independent set of G_1^L or of G_2^L or there is a partition $U=U_1\cup U_2$, where $|U_i|\geqslant 1$ and U_i is an independent set of G_i^L. In the last case, for all pairs $u\in U_1$ and $v\in U_2$, we have $L(u)\neq L(v)$. Therefore, the labels of U_1 and the labels of U_2 must be different:

$$L(U_1)\cap L(U_2)=\emptyset.$$

Hence, for each independent set U of $G^L=\bigcup(G_1,G_2)^L$, there is a set of labels $\emptyset\subset H\subset Z_m$, where:

(1) $U=U_1\cup U_2$,

(2) U_1 is an independent set of $G_1^L|_{K(H,G)}$,

(3) U_2 is an independent set of $G_2^L|_{K(Z_m\setminus H,G)}$.

If, conversely, U_1 is an independent set of $G_1^L|_{K(H,G)}$ and U_2 is an independent set of $G_2^L|_{K(Z_m\setminus H,G)}$, with $H\subset Z_m$, then $U=U_1\cup U_2$ is an independent set of G^L. □

Clearly, the labels $L(U)$ of an independent set U must be different:

$$|L(U)|=|U|.$$

Hence, only the possible maximum labelling sets must be determined for the subgraphs of the cograph recursively. Therefore, we get the following assertion.

Theorem 4.2. *For a transfer flow graph with constant number of labels $m\in\mathbb{N}$, the problem of finding a maximum independent set can be solved in polynomial time.*

For a general number m of labels we get the following complexity result.

Theorem 4.3. *The search for an independent set of size $k\in\mathbb{N}$ is* NP-*complete for transfer flow graphs.*

Proof. We give a transformation from 3-SAT (see [2]) to the independent set problem for transfer flow graphs. Let $\alpha = c_1 \wedge \cdots \wedge c_m$ be a formula in conjunctive normal form, with exactly three literals for each clause;

$$c_i = (y_{i1} \vee y_{i2} \vee y_{i3}),$$

with $y_{ij} \in \{x_1, \bar{x}_1, \ldots, x_n, \bar{x}_n\}$. Herefore, we define a transfer flow graph $G^L = (N, E)^L$ such that the formula α is satisfiable iff the graph G^L has an independent set of size m.

The nodes of the graph are given by

$$N = \{a_{ij} \mid 1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant 3\}$$

and the edges E of the cograph are generated by the following graph operations:

$$G = \bigcup(G_1, \ldots, G_n),$$

$$G_k = +((\{a_{ij} \mid y_{ij} = x_k\}, \emptyset), (\{a_{ij} \mid y_{ij} = \bar{x}_k\}, \emptyset)).$$

The labelling $L: N \to Z_m$ for this cograph is given by

$$L(a_{ij}) = i \quad \text{for all } 1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant 3.$$

The constructed graph G^L is the complement graph of a graph, constructed by Karp [8] for the clique problem. He has shown that the formula α is satisfiable iff there is a clique of size m in the complement of G^L. This implies the assertion of the theorem. □

5. Partition into cliques

Now we want to examine the problem of finding a partition into k cliques. Depending upon the recursive structure of a cograph, we get a recursive order to compute the minimum clique-partition number $\kappa(G^L)$.

Lemma 5.1. *Let G^L be a transfer flow graph, where $G = (N, E) = op(G_1, G_2)$ and $G_i = (N_i, E_i)$ are cographs and $L: N \to Z_m$ is a labelling.*

If $op = +$, we get

$$\kappa(G^L) = \max\{\kappa(G_1^L), \kappa(G_2^L)\}.$$

If $op = \cup$, we get

$$\kappa(G^L) = \min_{\emptyset \subset H \subset Z_m} \{\kappa(G_1^L|_{K(Z_m \setminus H, G)}) + \kappa(G_2^L|_{K(Z_m \setminus H, G)}) + |H|\}.$$

Proof. Let $C_1, \ldots, C_k$ be a partition into k cliques for $+(G_1, G_2)^L$. Each of these cliques C_i can be divided into two cliques C_i^1 and C_i^2, where C_i^j is a clique of G_j^L.

Altogether, we obtain a partition into at most k cliques for G_j^L. Conversely, for a partition for G_1^L into k cliques and a partition for G_2^L into l cliques, we get, by component union, a partition of G^L into $\max(k, l)$ cliques.

For the other case, let $C_1, \ldots, C_k$ be a clique partition for $\bigcup(G_1, G_2)^L$. Then C_i is a clique of G_j^L for one $j \in \{1, 2\}$ or we have the property $|L(C_i)| = 1$. Without loss of generality, it is given a set H with $\emptyset \subset H \subset Z_m$, where $C_1, \ldots, C_{|H|}$ are cliques with $|L(C_i)| = 1$ and

$$\bigcup_{i=1}^{|H|} L(C_i) = H,$$

$$\bigcup_{i=|H|+1}^{k} L(C_i) \cap H = \emptyset.$$

Consequently, this partition consists of $|H|$ cliques with $|L(C_i)| = 1$, a clique partition for $G_1^L|_{K(Z_m \setminus H)}$ and a clique partition for $G_2^L|_{K(Z_m \setminus H)}$. $\square$

By the recursive computation order of Lemma 5.1, we get the following complexity result.

Theorem 5.2. *If k or m is constant, the problem of finding a partition into k cliques can be solved in polynomial time for a transfer flow graph G^L with m labels.*

For general k and m, the clique partition problem is more difficult. An examination of the complexity gives the following result.

Theorem 5.3. *The problem of finding a partition into k cliques remains* NP*-complete for a transfer flow graph.*

Proof. We use a transformation of Garey and Johnson [2] from the exact cover by 3-sets problem to partition into cliques problem. Let $c_i = \{x_{i1}, x_{i2}, x_{i3}\}$ be a 3-set of a set X, with $X = \{x_1, \ldots, x_{3q}\}$ for each $1 \leqslant i \leqslant m$. An exact cover for X is a selection of sets from $\{c_1, \ldots, c_m\}$ such that each element of X occurs exactly once.

For an instance of the exact cover problem, we define a cograph $G = (N, E)$, with

$$N = X \cup \bigcup_{i=1}^{m} \{a_{ij} \mid 1 \leqslant j \leqslant 9\},$$

$$E = \bigcup_{i=1}^{m} E_i,$$

$$E_i = \left\{ \begin{array}{l} \{x_{i1}, a_{i1}\}, \{x_{i1}, a_{i2}\}, \{a_{i1}, a_{i2}\}, \\ \{x_{i2}, a_{i4}\}, \{x_{i2}, a_{i5}\}, \{a_{i4}, a_{i5}\}, \\ \{x_{i3}, a_{i7}\}, \{x_{i3}, a_{i8}\}, \{a_{i7}, a_{i8}\}, \\ \{a_{i3}, a_{i6}\}, \{a_{i3}, a_{i9}\}, \{a_{i6}, a_{i9}\} \end{array} \right\}$$

and a labelling $L: N \to Z_{3(m+q)}$, with

$$L(a_{ij}) = \lceil \tfrac{j}{3} \rceil + 3(i-1),$$

$$L(x_j) = 3m + j.$$

The generated graph G^L is equal to the graph constructed by Garey and Johnson [2]. For this, it was shown that there is an exact cover for X in q 3-sets iff there is a partition for G^L in $3m+q$ cliques. □

Corollary 5.4. *Let $(\mathscr{F}, M, typ, T)$ be a transfer digraph with a set Op of transfer operations. A minimum bus assignment can be determined in polynomial time if the number of modules M or the number of busses are constant. In the general case the bus-assignment problem is* NP*-complete.*

6. Coloring

Since the independent set problem is NP-complete, there is only a weaker assertion for the recursive structure for the colorings. Therefore, we consider for a partition into independent sets $U_1, \ldots, U_k$ their labelling sets $L(U_1), \ldots, L(U_k)$. With the assistance of these labelling sets, we have the following theorem.

Lemma 6.1. *Let G^L be a transfer flow graph, where $G=(N,E)=op(G_1,G_2)$ and $G_i=(N_i,E_i)$ are cographs and $L: N \to Z_m$ is a labelling.*

If $op = +$, we get:

> *There is a partition of G^L into k independent sets if there is, for an integer $h \in \{1, \ldots, k\}$, a partition of G_1^L into h independent sets and a partition of G_2^L into $k-h$ independent sets.*

If $op = \cup$, we get:

> *There is a partition of G^L into k independent sets with labelling sets $L_1, \ldots, L_k$ if there are decompositions $A_{i,1} \cup A_{i,2} = L_i$ for $1 \leqslant i \leqslant k$ (eventually, $A_{i,j} = \emptyset$) and if there are partitions of G_j^L into at most k independent sets with labelling sets $A_{1,j}, \ldots, A_{k,j}$ for $j \in \{1,2\}$.*

Proof. The proof, herefore, follows from the properties of the independent sets, which are given for a transfer flow graph in Lemma 4.1. □

In what follows, we consider vectors $x = (x_{\{1\}}, \ldots, x_{\{1,\ldots,m\}})$, where $x_U \in \mathbb{N}_0$ specifies the number of labelling sets $U \subset Z_m$. A vector is a solution if there is a coloring with these numbers of labelling sets.

If $G^L = (\{n\}, \emptyset)^L$ with labelling $L(n)$, each vector x, with $x_U = 1$ for exactly one set $U \subset Z_m$ with $L(n) \in U$, and $x_{U'} = 0$ otherwise, is a solution.

If $G^L = +(G_1, G_2)^L$ with solution sets L_i for G_i^L, then each vector $x + y$, with $x \in L_1$ and $y \in L_2$, is a solution of G^L.

In the case $G^L = \bigcup(G_1, G_2)^L$ with solution sets L_i for G_i^L, each vector z, which is given by an assignment of vectors $x \in L_1$ and $y \in L_2$ in the following bipartite graph, is a solution of G^L. The bipartite graph (N, E), with $N = N_1 \cup N_2$, is defined as follows:

$$N_i = \{(A, i) \mid \emptyset \subset A \subset \{1, \dots, m\}\},$$

$$E = \{((A, 1), (B, 2)) \mid (A, 1) \in N_1, (B, 1) \in N_2, A \cap B = \emptyset, A \cup B \neq \emptyset\}.$$

In Fig. 1, the bipartite graph for $m = 2$ is shown. Each mapping $f: E \to \mathbb{N}_0$ with conditions (1) and (2) generates a solution of G^L:

(1) $\sum_{U \mid U \cap V = \emptyset} f(((U, 1), (V, 2))) = y_V \ \forall \emptyset \neq V \subset \{1, \dots, m\}$,

(2) $\sum_{V \mid U \cap V = \emptyset} f(((U, 1), (V, 2))) = x_U \ \forall \emptyset \neq U \subset \{1, \dots, m\}$.

The vector $z = (z_{\{1\}}, \dots, z_{\{1, \dots, m\}})$, which is given by (3), is such a solution:

(3) $\displaystyle\sum_{U, V \mid P = U \cup V} f(((U, 1), (V, 2))) = z_P \ \forall \emptyset \neq P \subset \{1, \dots, m\}$.

Theorem 6.2. *The problem of finding a k-coloring in a transfer flow graph with constant number m of labels can be solved in polynomial time.*

Proof. We can generate all solution vectors $x = (x_{\{1\}}, \dots, x_{\{1, \dots, m\}})$, with

$$\sum_{A \mid \emptyset \neq A \subset \{1, \dots, m\}} x_A \leqslant k$$

and a corresponding coloring. Then each set of solutions for a subgraph of the cograph consists of at most k^{2^m} (therefore, polynomial) many elements. The computation

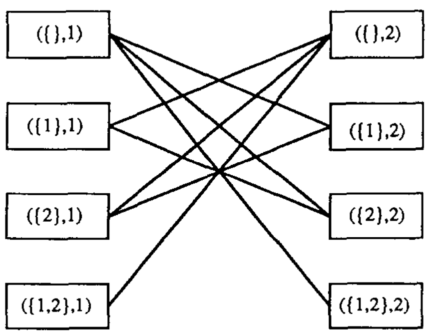

Fig. 1. The bipartite graph for $m = 2$.

of the solutions for $op = +$ is simple. Since the bipartite graph has a constant number of nodes, for each vector pair x, y, there are at most polynomial many feasible mappings $f: E \to \{0, \dots, k\}$. Therefore, the solutions for $op = \cup$ can be determined in polynomial many steps. $\square$

If m is not constant, we obtain the following assertion.

Theorem 6.3. *The question if there is a 3-coloring for a transfer flow graph remains* NP-*complete.*

Proof. We use a transformation from the 3-coloring problem with no vertex degree exceeding 4, which is NP-complete [2], to the coloring problem for a transfer flow graph. Let $G = (N, E)$ be a graph with $N = \{1, \dots, r\}$ and maximum degree 4. For each edge $\{i, k\} \in E$, with $i < k$, we define

$$j(i,k) = |\{\{i,k'\} \in E, k' \leqslant k\}|,$$

$$h(i,k) = |\{\{i',k\} \in E, i' \leqslant i\}|,$$

$$w(i,k) = |\{\{i',h\} \in E, i' < i, i' \leqslant h\}| + |\{\{i,k'\} \in E, i < k' \leqslant k\}|.$$

With the mapping w, we assign each edge a unique value between 1 and $|E|$. The mappings j and h specify which successor is k from i or which successor is i from k, respectively.

We substitute for each node $1, \dots, r$ a graph H_i and construct a transfer flow graph Q^L such that G is 3-colorable if and only if Q^L is 3-colorable.

For the node substitution, we use the graph H_i with 9 nodes, shown in Fig. 2. The graph H_i has 4 outlets, labelled by $y_{i1}, y_{i2}, y_{i3}, y_{i4}$. This graph satisfies the following conditions:

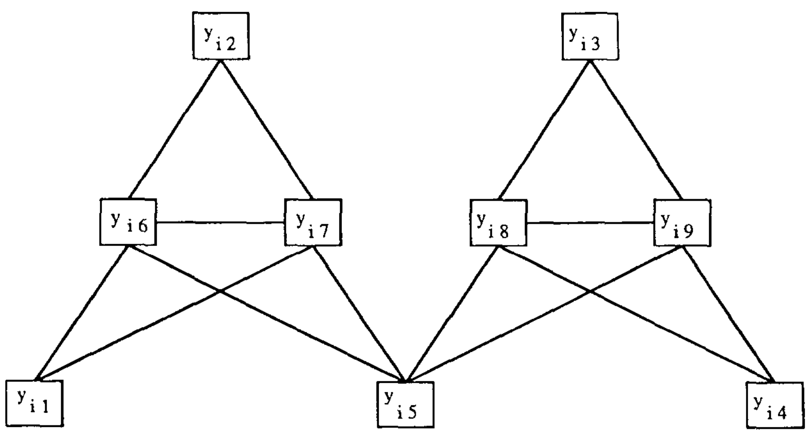

Fig. 2. The graph H_i.

(1) No node of H_i has degree exceeding 4.
(2) Each outlet has degree 2.
(3) H_i is 3-colorable but not 2-colorable and, for each 3-coloring of H_i, we have

$$f(y_{i1})=f(y_{i2})=f(y_{i3})=f(y_{i4})=f(y_{i5}).$$

(4) The graph K_i, which arises from H_i by deleting of the edges $\{y_{i5}, y_{i6}\}$ and $\{y_{i5}, y_{i7}\}$, is a cograph.

We define the cograph Q by

$$Q=\bigcup_{i=1}^{r} K_i,$$

with nodes

$$\{y_{ij} \mid 1 \leqslant i \leqslant r,\ 1 \leqslant j \leqslant 9\}.$$

Moreover, we choose a mapping L where, for each edge $\{i,k\} \in E$, with $i<k$,

$$L(y_{ij(i,k)})=L(y_{kh(i,k)})=w(i,k)$$

and, for each $i \in \{1, \ldots, r\}$,

$$L(y_{i5})=L(y_{i6})=L(y_{i7})=|E|+i.$$

All other nodes are assigned to different numbers between $|E|+r+1$ and $|E|+3r+\sum_{i=1}^{r} 4-|\Gamma(i)|$.

We have an edge $\{i,k\}$ in G iff the graphs H_i and H_k are connected in Q^L. Hence, G is 3-colorable iff Q^L is 3-colorable. □

7. Results

Altogether, for this graph class, the transfer flow graphs, we get the complexity results summarized in the Table 1. The entry P means that the corresponding problem can be solved in polynomial time and the entry NPc that the problem is NP-complete.

Table 1
Complexity results

Problem	General	m constant	k constant
Clique (k)	P	P	P
Independent set (k)	NPc	P	P
Partition into cliques (k)	NPc	P	P
Coloring (k)	NPc	P	NPc

References

[1] C. Berge, Graphs and Hypergraphs (North-Holland, Amsterdam, 1970).
[2] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness (Freeman, San Francisco, 1979).
[3] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs (Academic Press, New York, 1980).
[4] M. Grötschel, A. Schrijver and L. Lovász, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica 1 (1981) 169–197.
[5] F. Harary, Graph Theory (Addison-Wesley, Reading, 1969).
[6] M.S. Hecht, Flow Analysis of Computer Programs (Elsevier, New York, 1977).
[7] D.S. Johnson, The NP-completeness column: on ongoing guide, J. Algorithms 6 (1985) 434–451.
[8] R.M. Karp, Reducibility among combinatorial problems, in: R.E. Miller and J.W. Thatcher, eds., Complexity of Computer Computations (Plenum, New York, 1972) 85–103.
[9] L. Lovász, Perfect graphs, in: L.W. Beineke and R.J. Wilson, eds., Selected Topics in Graph Theory 2, (Academic Press, New York, 1983) 55–87.
[10] D. Seinsche, On a property of the class of n-colorable graphs, J. Combin. Theory 16 (1974) 191–193.
[11] H.C. Torng and N.C. Wilhelm, The optimal interconnection of circuit modules in microprocessor and digital system design, IEEE Trans. Comput. 5 (1977) 450–457.
[12] C.J. Tseng and D.P. Siewiorek, The modelling and synthesis of bus systems, 18th Design Automation Conf., 1981, 471–478.